In the Claims:

Please cancel claims 15 and 24-38. Please amend claims 16 and 23. Please add new claims 39-51. The claims are as follows:

1-14. (Canceled)

15. (Canceled)

16. (Currently amended) ~~The method of claim 15~~ A method for controlling bottlenecks in an information system that includes N application servers and a database server, wherein N is at least 2, wherein each application server is adapted to execute at least one application program for processing a transaction received by each application server from a terminal, wherein the database server is adapted to access a database based on a request received from any application server of the N application servers, and wherein the method comprises:

monitoring a processing time required for each application program to process the transaction received by each application server;

detecting a bottleneck relating to usage of at least one resource, wherein each resource of the at least one resource is independently selected from the group consisting of a resource of at least one application server of the N application servers, a resource related to input to the transaction, a resource of the database server, and a resource related to the transaction, wherein said detecting is responsive to said monitoring having determined that the processing time for processing the transaction by I application servers of the N application servers is not within a predesignated permissible processing,

wherein M denotes a predesignated threshold number of application servers, wherein said detecting the bottleneck relating to usage of at least one resource comprises identifying the at least one resource, and wherein said identifying the at least one resource comprises independently identifying each resource of the at least one resource as being:

said resource of at least one application server of the N application servers if I is at least 1 and does not exceed M and if a processing time for processing another type of transaction by any application server of the N application servers is not within the predesignated permissible processing time range;

said resource related to input to the transaction if I is at least 1 and does not exceed M and if a processing time for processing another type of transaction by any application server of the N application servers is within the predesignated permissible processing time range;

said resource of the database server if I exceeds M and if a processing time for processing another type of transaction by any application server of the N application servers is not within the predesignated permissible processing time range;

said resource ~~resource~~ related to the transaction if I exceeds M and if a processing time for processing another type of transaction by any application server of the N application servers is within the predesignated permissible processing time range.

17. (Previously presented) The method of claim 16, wherein the method further comprises monitoring processing loads imposed on: resources of the N application servers, resources of the database server, and resources related to the transaction, and wherein said identifying each resource of the at least one resource comprises determining from the monitored processing loads that a high load specific to each resource of the at least one resource is imposed on each resource of the at least one resource

18. (Previously presented) The method of claim 17, wherein said determining that a high load is imposed on each resource of the at least one resource comprises determining that a predesignated detection condition has occurred for each resource of the at least one resource a predesignated number of times, and wherein the predesignated detection condition is that a predesignated usage parameter specific to each resource of the at least one resource is in a predesignated load range.

19. (Previously presented) The method of claim 17, wherein said removing the detected bottleneck comprises eliminating the high load imposed on each resource of the at least one resource.

20. (Previously presented) The method of claim 19, wherein said eliminating comprises executing in a predesignated sequence specific to each resource of the at least one resource as many of one or more predesignated load control processes as is necessary to eliminate the high load imposed on each resource of the at least one resource.

21. (Previously presented) The method of claim 20, wherein a first resource of the at least one resource is a resource of a first application server of the N application servers, wherein said executing the predesignated sequence specific to the first resource comprises reducing an application program multiplicity of the first application server, and wherein said application program multiplicity on the first application server is defined as a maximum number of application programs to be executed concurrently on the first application server with respect to a plurality of transactions of the same type that were received by the first application server at the same time.

22. (Previously presented) The method of claim 20, wherein a first resource of the at least one resource is a resource of the database server, and wherein said executing the predesignated sequence specific to the first resource comprises reducing a priority level of a process for accessing the database.

23. (Currently amended) The method of claim [[15]] 16, wherein an upper limiting processing time of the predesignated permissible processing time range is one standard deviation higher than an average processing time per transaction processed during peak processing loads during a predesignated period of time.

24-38. (Canceled)

39. (New) The method of claim 16, wherein I is at least 1 and does not exceed M, and wherein the processing time for processing another type of transaction by any application server of the N application servers is not within the predesignated permissible processing time range.

40. (New) The method of claim 39, wherein I is at least 2 and is less than M.

41. (New) The method of claim 39, wherein I is at least 2 and is equal to M.

42. (New) The method of claim 16, wherein I is at least 1 and does not exceed M, and wherein the processing time for processing another type of transaction by any application server of the N application servers is within the predesignated permissible processing time range.

43. (New) The method of claim 42, wherein I is at least 2 and is less than M..

44. (New) The method of claim 42, wherein I is at least 2 and is equal to M.

45. (New) The method of claim 16, wherein I exceeds M, and wherein the processing time for processing another type of transaction by any application server of the N application servers is not within the predesignated permissible processing time range.

46. (New) The method of claim 45, wherein I is at least 2 and is less than M..

47. (New) The method of claim 45, wherein I is at least 2 and is equal to M.

48. (New) The method of claim 16, wherein I exceeds M, and processing time for processing another type of transaction by any application server of the N application servers is within the predesignated permissible processing time range.

49. (New) The method of claim 48, wherein I is at least 2 and is less than M..

50. (New) The method of claim 48, wherein I is at least 2 and is equal to M.

51. (New) A method for controlling bottlenecks in an information system that includes N application servers and a database server, wherein N is at least 2, wherein each application server is adapted to execute at least one application program for processing a transaction received by each application server from a terminal, wherein the database server is adapted to access a database based on a request received from any application server of the N application servers, and wherein the method comprises:

monitoring a processing time required for each application program to process the transaction received by each application server;

detecting a bottleneck relating to usage of at least one resource, wherein each resource of the at least one resource is independently selected from the group consisting of a resource of at least one application server of the N application servers, a resource related to input to the transaction, a resource of the database server, and a resource related to the transaction, wherein said detecting is responsive to said monitoring having determined that the processing time for processing the transaction by I application servers of the N application servers is not within a predesignated permissible processing,

wherein M denotes a predesignated threshold number of application servers, wherein said detecting the bottleneck relating to usage of at least one resource comprises identifying the at least one resource, and wherein said identifying the at least one resource comprises independently identifying each resource of the at least one resource as being:

said resource of at least one application server of the N application servers if I is at least 1 and does not exceed M and if a processing time for processing another type of transaction by any application server of the N application servers is not within the predesignated permissible processing time range;

said resource related to input to the transaction if I is at least 1 and does not exceed M and if a processing time for processing another type of transaction by any application server of the N application servers is within the predesignated permissible processing time range;

said resource of the database server if I exceeds M and if a processing time for processing another type of transaction by any application server of the N application servers is not within the predesignated permissible processing time range;

said resource related to the transaction if I exceeds M and if a processing time for processing another type of transaction by any application server of the N application servers is within the predesignated permissible processing time range.

wherein the method further comprises monitoring processing loads imposed on: resources of the N application servers, resources of the database server, and resources related to the transaction, and wherein said identifying each resource of the at least one resource comprises determining from the monitored processing loads that a high load specific to each resource of the at least one resource is imposed on each resource of the at least one resource,

wherein said determining that a high load is imposed on each resource of the at least one resource comprises determining that a predesignated detection condition has occurred for each resource of the at least one resource a predesignated number of times, and wherein the predesignated detection condition is that a predesignated usage parameter specific to each resource of the at least one resource is in a predesignated load range,

wherein said removing the detected bottleneck comprises eliminating the high load imposed on each resource of the at least one resource,

wherein said eliminating comprises executing in a predesignated sequence specific to each resource of the at least one resource as many of one or more predesignated load control processes as is necessary to eliminate the high load imposed on each resource of the at least one resource,

wherein a first resource of the at least one resource is a resource of a first application server of the N application servers, wherein said executing the predesignated sequence specific to the first resource comprises reducing an application program multiplicity of the first application server, and wherein said application program multiplicity on the first application server is defined as a maximum number of application programs to be executed concurrently on the first application server with respect to a plurality of transactions of the same type that were received by the first application server at the same time,

wherein an upper limiting processing time of the predesignated permissible processing time range is one standard deviation higher than an average processing time per transaction processed during peak processing loads during a predesignated period of time.